

BAB II

TINJAUAN PUSTAKA

II.1. Pengertian Sistem

Secara sederhana, suatu sistem dapat diartikan sebagai suatu kumpulan himpunan dari unsur, komponen, atau variabel yang terorganisir, saling berinteraksi, saling tergantung satu sama lain, dan terpadu. Teori sistem melahirkan konsep-konsep futuristik. Salah satu konsep yang terkenal adalah konsep sibernetika (*cybernetics*). Konsep kajian ilmiah ini terutama berkaitan dengan upaya menerapkan berbagai disiplin ilmu, yaitu ilmu perilaku, fisika, biologi dan teknik. Oleh karena itu, sibernetika biasanya berkaitan dengan usaha-usaha otomasi tugas-tugas yang dilakukan oleh manusia sehingga studi tentang robotika, kecerdasan buatan (*Artificial Intelligence*), dan lain adalah masukan (*input*), pengolahan (*processing*), dan keluaran (*output*).

Konsep lain yang terkandung di dalam definisi tentang sistem adalah konsep sinergi. Konsep ini mengandaikan bahwa di dalam suatu sistem. *Output* dari suatu organisasi diharapkan lebih besar dari pada *output* individual atau *output* masing-masing bagian.

Sebuah *system* terdiri atas bagian-bagian atau komponen yang terpadu untuk satu tujuan. Model dasar dari bentuk sistem ini adalah adanya masukan, pengolahan, dan keluaran. Akan tetapi, sistem ini dapat dikembangkan hingga menyetakan media penyimpanan. Sistem dapat terbuka dan tertutup. Sistem informasi biasanya adalah sistem terbuka, yang berarti bahwa sistem tersebut

dapat menerima berbagai masukan dari lingkungan sekitarnya. (Tata Sutabri 2012 : 3-4)

II.1.1 Karakteristik Sistem

Adapun karakteristik yang mencirikan suatu sistem, yaitu :

1. **Komponen Sistem (*Component*)**

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem tersebut dapat berupa suatu bentuk subsistem.

2. **Batasan Sistem (*Boundary*)**

Ruang lingkup sistem merupakan daerah yang membatasi antara sistem dengan sistem yang lain atau sistem dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai satu kesatuan yang tidak dapat dipisahkan.

3. **Lingkungan Luar Sistem**

Bentuk apapun yang ada diluar ruang lingkup atau batasan sistem yang mempengaruhi operasi sistem tersebut disebut lingkungan luar sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat merugikan sistem tersebut.

4. **Penghubung Sistem (*Interface*)**

Media yang menghubungkan sistem dengan sub sistem lain disebut penghubung sistem (*Interface*).

5. **Masukan Sistem (*Input*)**

Energi yang dimasukkan ke dalam sistem disebut masukan sistem, yang dapat berupa pemeliharaan (*maintenance input*) dan sinyal (*signal input*).

6. Keluaran Sistem (*Output*)

Hasil energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna. Keluaran ini merupakan masukan bagi subsistem yang lain seperti sistem informasi. Keluaran yang dihasilkan adalah informasi. Informasi ini digunakan sebagai masukan untuk pengambilan keputusan atau hal-hal lain yang menjadi input bagi subsistem yang lain.

7. Pengolahan Sistem (*Procces*)

Suatu sistem dapat mempunyai suatu proses yang akan mengubah masukan menjadi keluaran.

8. Sasaran Sistem (*Objective*)

Suatu sistem memiliki tujuan dan sasaran yang pasti dan bersifat deterministik. Kalau suatu sistem tidak memiliki sasaran maka operasi sistem tidak ada gunanya. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuan yang telah direncanakan. (Tata Sutabri; 2012 : 12-13)

II.1.2 Pengertian Informasi

Informasi adalah data yang telah diklasifikasikan atau diinterpretasi untuk digunakan dalam proses pengambilan keputusan. Sistem pengolahan informasi akan mengolah data menjadi informasi atau mengolah data dari bentuk tak berguna menjadi berguna bagi yang menerimanya. Nilai informasi berhubungan

dengan keputusan. Bila tidak ada pilihan atau keputusan maka informasi tidak diperlukan. Keputusan dapat berkisar dari keputusan berulang sederhana sampai keputusan strategis jangka panjang. (Tata Sutabri; 2012 : 18)

Fungsi utama informasi adalah menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi. Informasi yang disampaikan kepada pemakai mungkin merupakan hasil dari data yang dimasukkan ke dalam pengolahan. Akan tetapi dalam kebanyakan pengambilan keputusan yang kompleks, informasi hanya dapat menambah kemungkinan kepastian atau mengurangi bermacam-macam pilihan. (Tata Sutabri; 2012 : 19)

II.1.3 Pengertian Sistem Informasi

Sistem informasi bukan merupakan hal yang baru. Yang baru adalah komputerisasinya. Sebelum ada komputer, teknik penyaluran informasi yang memungkinkan manajer merencanakan serta mengendalikan operasi.

Sistem informasi adalah suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat *manajerial* dengan kegiatan startegi dari suatu orgnaisai untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan. (Tata Sutabri; 2012 : 36)

II.1.4 Komponen sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan (*Building Block*) yang terdiri dari :

1. Blok Masukan (*Input Block*)

Input mewakili data yang masuk kedalam sistem informasi.

2. Blok Model (*Model Block*)

Blok ini terdiri dari kombinasi prosedur, logika, dan model matematik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang diinginkan.

3. Blok Keluaran (*Output Block*)

Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi (*Technology Block*)

Teknologi merupakan “*tool box*” dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran, dan membantu pengendalian dari sistem secara keseluruhan.

5. Blok Basis Data

Basis data (*database*) merupakan kumpulan data yang saling berkaitan berhubungan satu sama lain, tersimpan diperangkat keras computer dan menggunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali (*Control Block*)

Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah

ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi. (Tata Sutabri ; 2012 : 36-37)

II.2 Sistem Informasi Geografis

Istilah “*Geografis*“ merupakan bagian dari spasial (keruangan). Penggunaan kata “*Geografis*” mengandung pengertian suatu persoalan atau hal mengenai (wilayah di permukaan) bumi, baik dua dimensi atau tiga dimensi, dengan demikian istilah ”informasi geografis” mengandung pengertian informasi mengenai tempat- tempat yang terletak di permukaan bumi, pengetahuan mengenai posisi di mana suatu objek terletak di bumi, atau informasi mengenai keterangan-keterangan (atribut) objek penting yang terdapat di permukaan bumi yang posisinya diberikan atau diketahui.

SIG juga merupakan sejenis perangkat lunak, perangkat keras (manusia, prosedur, basis data, dan fasilitas jaringan komunikasi) yang dapat digunakan atau memfasilitasi proses pemasukan, penyimpanan, manipulasi, menampilkan, dan keluaran data. (Eddy Prahasta 2009 : 109-110)

II.2.I Subsistem SIG

SIG dapat diuraikan menjadi beberapa subsistem yaitu :

1. *Data Input*

Bertugas untuk mengumpulkan, mempersiapkan, dan menyimpan data spasial dan atributnya dari berbagai sumber.

2. *Data Output*

Bertugas untuk menampilkan atau menghasilkan keluaran (termasuk

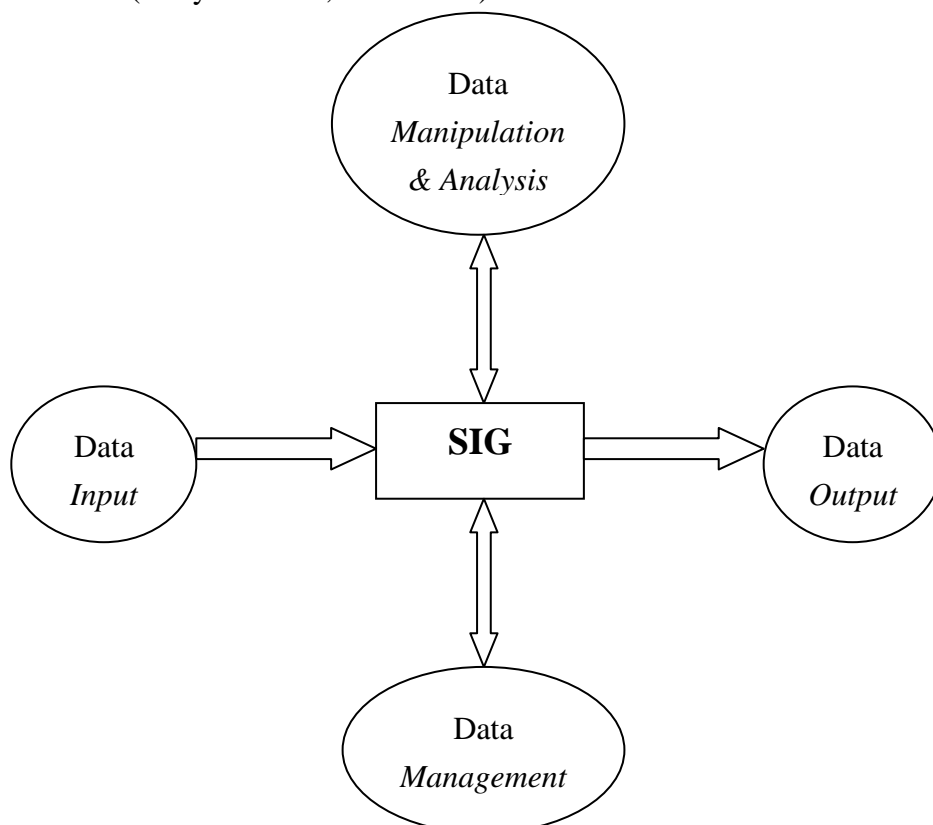
mengekspornya ke format yang di kehendaki seluruh atau sebagian basis data spasial) baik dalam bentuk *softcopy*, maupun *hardcopy* seperti halnya tabel, grafik, *report*, peta, dan lain sebagainya.

3. *Data Management*

Bertugas mengorganisasikan baik data spasial maupun tabel-tabel atribut terkait ke dalam sebuah sistem basis data sedemikian rupa hingga mudah dipanggil kembali atau di *update* dan di *edit*.

4. *Data Manipulation dan Analysis*

Bertugas menentukan informasi-informasi yang dapat dihasilkan oleh SIG. (Eddy Prahasta; 2009 : 118)



Gambar II.1 : Subsistem Informasi
(Sumber : Eddy Prahasta; 2009 : 1)

II.2.2 Komponen SIG

SIG sebagai sistem terdiri dari beberapa komponen dengan karakteristiknya yaitu :

1. Perangkat Keras

Perangkat keras yang sering digunakan untuk aplikasi SIG adalah komputer, *mouse*, monitor (*plus VGA-card grafik*) yang beresolusi tinggi, *digitizer*, *printer*, *plotter*, *receiver GPS* dan *scanner*.

2. Perangkat Lunak

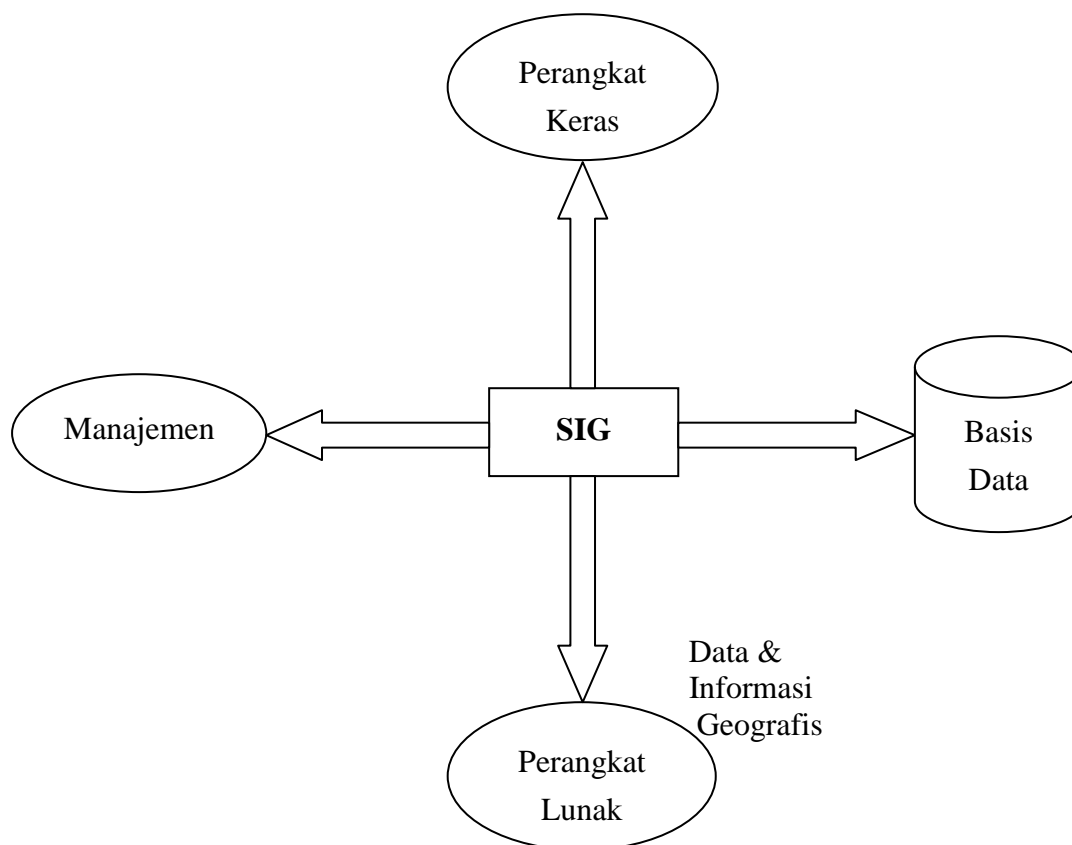
SIG bisa merupakan sistem perangkat lunak yang tersusun secara modular di mana sistem basis datanya memegang peranan kunci.

3. Data dan Informasi Geografis

SIG dapat mengumpulkan dan menyimpan data atau informasi yang diperlukan baik secara tidak langsung (dengan cara meng-*import*-nya dari format-format perangkat lunak SIG yang lain) maupun secara langsung dengan cara melakukan digitasi data spasialnya (digitasi *on-screen* atau *head-ups*) diatas tampilan layar monitor atau manual dengan menggunakan (*digitizer*) dari peta analog dan kemudian memasukkan data atributnya dari tabel-tabel atau laporan dengan menggunakan *keyboard*.

4. Manajemen

Suatu proyek SIG akan berhasil jika dikelola dengan baik dan dikerjakan oleh orang-orang memiliki keahlian (kesesuaian dengan *job-description* yang bersangkutan) yang tepat pada semua tingkatan. (Eddy Prahasta; 2009 : 120-121)



Gambar II.2 : Komponen-komponen SIG
 (Sumber : Eddy Prahasta; 2009 : 121)

II.3 ArcView

Arcview merupakan salah satu perangkat lunak (*tool*) SIG dan pemetaan yang dikembangkan oleh *ESRI* (*Environmental Systems Research instite*). Perangkat lunak ini memberikan fasilitas teknis yang berkaitan dengan pengolahan data spasial. Kemampuan grafis yang baik dan kemampuan teknis dalam pengolahan data spasial tersebut memberikan kekuatan secara nyata pada *ArcView* untuk melakukan analisis spasial. Kekuatan analisis spasial inilah yang pada akhirnya menjadikan *ArcView* banyak diterapkan dalam berbagai pekerjaan seperti analisis pemasaran, perencanaan wilayah dan tata ruang, sisitem informasi persil, pengendalian dampak lingkungan bahkan untuk keperluan militer.

Avenue adalah sebuah skrip atau bahasa pemrograman berorientasi objek (*Object Oriented Programming*). *Avenue* dapat dibentuk sebuah interface baru pada *Arcview*, otomatisasi pekerjaan-pekerjaan yang bersifat berulang (*repetitif*), ataupun membuat sebuah alur analisis spasial khusus yang belum terdapat pada *Arcview* tersebut. (Eko Budiyanto; 2010 : 177)

II.4 PHP

PHP merupakan bahasa standart yang digunakan dalam dunia *web site*. *PHP* adalah bahasa program yang berbentuk script yang diletakkan di dalam *server web*. Jika kita lihat dari sejarah, mulanya *PHP* diciptakan dari ide Rasmus Ledrof yang membuat sebuah script perl. Script tersebut sebenarnya dimaksudkan untuk digunakan sebagai program untuk dirinya sendiri. Akan tetapi, kemudian dikembangkan lagi sehingga menjadi sebuah bahasa yang disebut “Personal Home Page”. Inilah awal mula munculnya *PHP*. (Bunafit Nugroho ; 2004 : 141)

Ada beberapa cara menuliskan script *PHP* yaitu :

1. `<? Skript PHP ada disini ?>` atau
2. `<? php Skript PHP anda disini ?>` atau
3. `<% Skript PHP anda disini %>` atau
4. `<SCRIPT language="php"> Skript PHP anda disini </SCRIPT>`

Jadi, semua kata dan script yang diletakkan pada daerah script akan dianggap sebagai perintah *PHP* sehingga jika terjadi kesalahan atau kata-katanya tidak sesuai dengan program akan dianggap salah dan mengakibatkan program yang kita buat *error*. (Bunafit Nugroho ; 2004 : 144)

II.5 Database MySQL

MySQL (My Structure Query Language) atau yang biasa dibaca “mai-se-kuel” adalah sebuah program pembuat database yang bersifat *open source*, artinya siapa saja boleh menggunakan dan tidak dicekal. Saat kita mendengar *open source*, kita ingat dengan sistem operasi handal keturunan *Unix*.

Sebagai sebuah program penghasil database, MySQL tidak dapat berjalan sendiri tanpa adanya sebuah aplikasi lain (*interface*). *MySQL* dapat didukung oleh hampir semua program aplikasi baik yang *open source* seperti *PHP* maupun yang tidak, yang ada *platform windows* seperti *Visual Basic, Delphi*, dan lainnya. (Bunafit Nugroho ; 2004 : 29-30)

II.6 Sistem Basis Data

Basis data adalah kumpulan data yang saling berhubungan yang disimpan/diorganisasi secara bersama, dalam bentuk sedemikian rupa, dan tanpa redundansi (pengulangan) tidak perlu supaya dapat dimanfaatkan kembali dengan cepat dan mudah untuk memenuhi berbagai kebutuhan. (Ema Utami, Anggit Dwi Hartanto ; 2012 : 3)

Sistem basis data dapat terbagi dalam beberapa komponen, yaitu :

1. Data

Merupakan informasi yang disimpan dalam suatu struktur tertentu yang terintegrasi.

2. *Hardware*

Merupakan perangkat keras berupa komputer dengan media penyimpanan sekunder yang digunakan untuk menyimpan data karena pada umumnya

basis data memiliki ukuran yang besar.

3. Sistem Operasi

Program yang mengaktifkan/memfungsikan sistem komputer, mengendalikan seluruh sumber daya dalam komputer dan melakukan operasi-operasi dasar dalam komputer yang meliputi operasi *Input Output* (IO), pengelolaan *file* dan sebagainya.

4. Basis Data

Basis data sebagai inti dari sistem basis data. Basis data menyimpan data serta struktur sistem basis data baik untuk entitas maupun objek-objeknya secara detail.

5. *Database Management System (DBMS)*

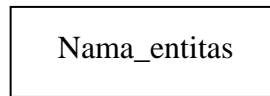
Merupakan perangkat lunak yang digunakan untuk melakukan pengelolaan basis data. (Erma Utami dan Anggit Dwi Hartono ; 2012 : 4)

II.6.1 *Entity Relational Diagram (ERD)*

ERD merupakan suatu diagram untuk menggambarkan desain konseptual dari model konseptual suatu basis data relasional. ERD juga merupakan gambaran yang menghubungkan antara objek satu dengan objek yang lain dalam dunia nyata. Bisa dikatakan bahwa bahan yang digunakan untuk membuat ERD adalah dari objek di dunia nyata. Secara umum ERD terdiri dari 3 komponen, yaitu :

1. Entitas (*Entity*)

Merupakan suatu “objek nyata” yang mampu dibedakan dengan objek yang lain. Objek tersebut dapat berupa orang benda ataupun hal yang lainnya. Penggambaran entitas dalam ERD seperti pada gambar II.3

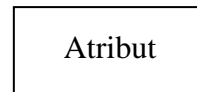


Gambar II.3 : Entitas

(Sumber : Ema Utami dan Anggit Dwi Hartanto ; 2012 : 19)

2. Atribut (*Attribute*)

Merupakan semua informasi yang berkaitan dengan entitas. Di dalam dunia pemograman, atribut adalah properti dari suatu objek. Penggambaran atribut dalam ERD seperti pada gambar II.4

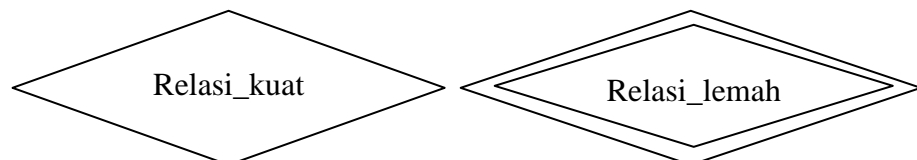


Gambar II.4 : Atribut

(Sumber : Ema Utami dan Anggit Dwi Hartanto ; 2012 : 20)

3. Relasi (*Relationship*)

Belah ketupat merupakan penggambaran hubungan (relasi) anatr entitas atau sering disebut kerelasiaan. Ada dua macam penggambaran relasi, yakni relasi kuat dan relasi lemah. Relasi kuat biasanya untuk menghubungkan antar entitas kuat, sedangkan realsi lemaha untuk menghubungkan antar entitas kuat dengan entitas lemah. Penggambaran kerelasiaan seperti gambar II.5



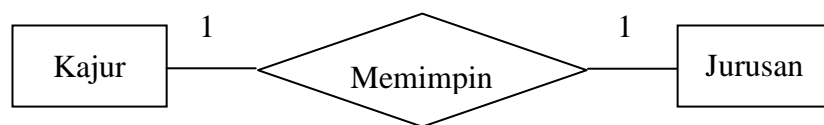
Gambar II.5 : Kerelasiaan

(Sumber : Ema Utami dan Anggit Dwi Hartanto ; 2012 : 24)

II.6.2 Derajat Kardinalitas

Merupakan penjelasan dari tingkat hubungan antar entitas. Ukuran derajat kardinalitas dibagi menjadi tiga macam, yaitu :

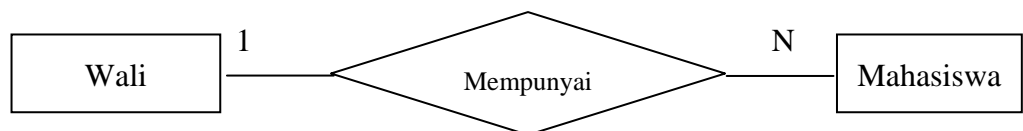
1. 1-1 (*one-to-one*), misalnya seorang ketua jurusan hanya memimpin satu jurusan, begitu juga sebaliknya satu jurusan hanya dipimpin seorang ketua jurusan, seperti gambar II.6



Gambar II.6 : 1-1 (*one-to-one*)

(Sumber : Ema Utami dan Anggit Dwi Hartanto ; 2012 : 24)

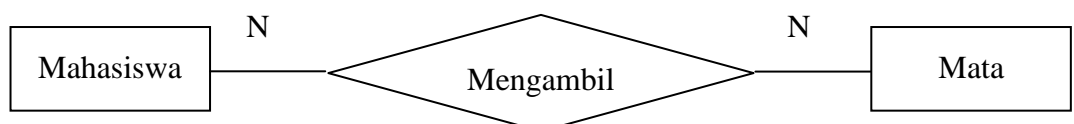
2. 1-N (*one-to-many*) atau N-1 (*many-to-one*), misalnya seorang mahasiswa hanya mempunyai seorang wali, tetapi seorang wali bisa menjadi wali banyak mahasiswa, seperti gambar II.7



Gambar II.7 : 1-N (*one-to-many*)

(Sumber : Ema Utami dan Anggit Dwi Hartanto ; 2012 : 25)

3. N-N (*many-to-many*), misalnya seorang mahasiswa bisa mengambil banyak mata kuliah, begitu juga sebaliknya satu mata kuliah bisa diambil oleh banyak, seperti gambar II.8



Gambar II.8 : N-N (*many-to-many*)

(Sumber : Ema Utami dan Anggit Dwi Hartanto ; 2012 : 25)

II.6.3 Normalisasi

Merupakan proses pengelompokan elemen data menjadi tabel yang menunjukkan entitas sekaligus relasinya. Tujuan dari normalisasi adalah mengurangi kemungkinan terjadinya anomali yang terjadi dalam basis data. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 40).

Adapun bentuk-bentuk normalisasi, yaitu :

1. Bentuk Normalisasi Pertama (1NF)

Suatu tael dikatakan dalam bentuk normal pertama apabila :

- a. Tidak ada baris data yang terduplikat atau berulang dalam tabel.
- b. Setiap sel memiliki nilai tunggal, artinya tidak ada perulangan *group* atau *array*.
- c. Data dalam kolom (atribut atau *field*) memiliki tipe data yang sejenis.

2. Bentuk Normalisasi Kedua (2NF)

Tabel dalam keadaan 2NF apabila tabel sudah dalam keadaan 1NF dan semua atribut yang bukan kunci bergantung pada semua kunci dalam tabel. 2NF bertujuan untuk menghilangkan ketergantungan parsial.

3. Bentuk Normalisasi Ketiga (3NF)

Tabel dalam keadaan 2NF apabila tabel dalam keadaan 2NF dan dalam tabel tersebut tidak ada ketergantungan transitif. Artinya sebuah *field* dapat menjadi atribut biasa pada suatu relasi tetapi menjadi kunci pada relasi lain. Setiap atribut yang bukan kunci haruslah bergantung hanya pada *primary key*.

4. Bentuk Normalisasi *Boyce-Codd* (BCNF)

Tabel dalam keadaan 3NF dan setiap determinan merupakan kunci kandidat. Determinan adalah suatu atribut/*field* atau gabungan atribut dimana beberapa atribut lain bergantung pada atribut tersebut.

5. Bentuk Normalisasi Keempat (4NF)

Tabel dalam keadaan BCNF dan tidak ada ketergantungan *multi value*.

6. Bentuk Normalisasi Kelima (5NF)

Tabel dalam keadaan 4NF dan setiap ketergantungan *join* dalam tabel merupakan akibat dari kunci kandidat tabel.

7. Bentuk Normalisasi *Domain-Key* (DKNF)

Tabel dikatakan dalam keadaan DKNF jika setiap *constraint* tabel merupakan akibat dari definisi kunci-kunci dan domain. (Ema Utami dan Anggit Dwi Hartanto ; 2012 : 41-42)

II.7 Dreamweaver

Dreamweaver adalah editor yang komplit yang dapat digunakan untuk membuat animasi sederhana yang berbentuk layer. Dengan adanya program ini kita tidak akan susah-susah mengetik script-script format HTML, PHP, ASP maupun bentuk program lainnya. (Bunafit Nugroho ; 2004 : 91)

II.8 Unified Modelling Language (UML)

UML singkatan dari *Unified Modelling Language* yang berarti bahasa permodelan standar. UML diaplikasikan untuk maksud tertentu, biasanya antara lain untuk :

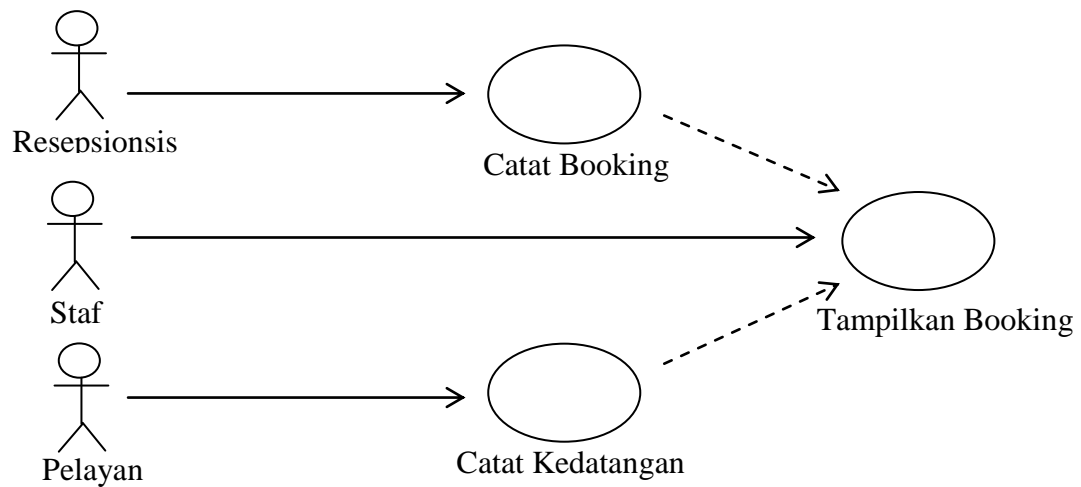
1. Merancang perangkat lunak
2. Sarana Komunikasi antara perangkat lunak dengan proses bisnis.
3. Menjabarkan sistem secara rinci untuk analisa dan mencari apa yang diperlukan sistem.
4. Mendokumentasi sistem yang ada, proses-proses dan organisasinya.

Blok pembangun utama *UML* adalah diagram. Beberapa ada diagram yang rinci (*jenis timing diagram*) dan lainnya ada yang bersifat umum (misalnya *diagram kelas*). Intinya, *UML* merupakan alat komunikasi yang konsisten dalam *mensupport* para pengembang sistem saat ini. (Prabowo Pudjo Widodo dan Herlawati ; 2011 : 6-7)

II.8.1. Use Case Diagram

Use Case adalah deskripsi fungsi dari sebuah sistem dari perspektif pengguna. *Use Case* bekerja dengan cara mendeskripsikan tipikal interaksi antar *user* (pengguna) sebuah sistem dan sistemnya sendiri melalui sebuah cerita bagaimana sebuah sistem dipakai. Urutan langkah-langkah yang menerangkan antara pengguna dan sistem disebut *scenario*.

Dalam pembicaraan tentang *use case*, pengguna biasanya disebut dengan *actor*. *Actor* adalah sebuah peran yang bisa dimainkan oleh pengguna dalam interaksinya dengan sistem. Contoh *use case diagram* dapat dilihat pada gambar II.9. menunjukkan ringkasan *use case* dari hasil diatas . (Munawar ; 2005 : 63)

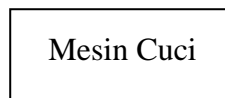


Gambar II.9 : Contoh Use Case Diagram

(Sumber : Munawar ; 2005 : 68)

II.8.2 Class Diagram

Class, dalam notasi *UML* digambarkan dengan kotak. Nama *class* menggunakan huruf besar di awal kalimatnya dan diletakkan diatas kotak. Bila *class* mempunyai nama yang terdiri dari 2 suku kata atau lebih, maka semua suku kata menggunakan huruf besar. Contoh *notasi class* di *UML* dapat dilihat gambar II.10. sebagai berikut :



Gamabar II.10 : Contoh Notasi class di UML

(Sumber : Munawar ; 2005 : 35)

Attribute adalah *property* dari sebuah *class*. *Attribute* ini melukiskan batas nilai yang mungkin ada pada obyek dari *class*. Sebuah *class* mungkin mempunyai nol atau lebih *attribute*. Secara konvensi, jika nama *attribute* mengandung lebih dari satu suku kata maka semua suku kata digabungkan dengan suku kata pertama

menggunakan huruf kecil dan awal suku kata berikutnya menggunakan huruf besar. Contoh *attribute* dapat dilihat pada gambar II.11. menunjukkan ringkasan *attribute* dari hasil diatas

Mesin Cuci
Merek
Model
NoSeri
Kapasitas

Gambar II.11 : Class dan attribute-attribute nya

(Sumber : Munawar ; 2005 : 35-36)

Operation adalah sesuatu yang bisa dilakukan oleh sebuah *class* atau yang anda (atau *class* yang lain) dapat dilakukan untuk sebuah *class*. Seperti halnya *attribute*, nama *operation* juga menggunakan huruf kecil semua jika terdiri dari suku kata. Akan tetapi jika lebih dari suku kata, maka semua kata digabungkan dengan suku kata pertama huruf kecil dan huruf awal tiap suku berikutnya dengan huruf besar. . Contoh *operation* dapat dilihat pada gambar II.12. menunjukkan ringkasan *operation* dari hasil diatas

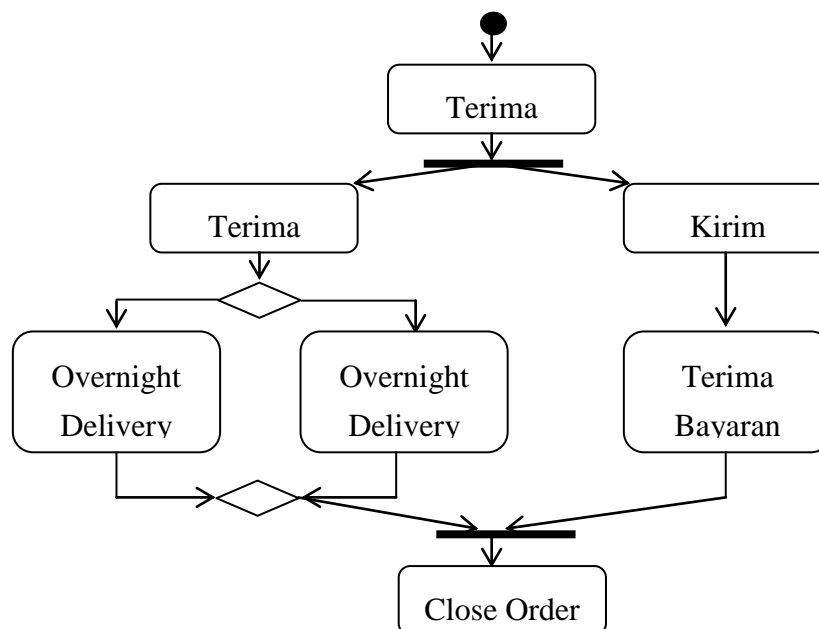
Mesin Cuci
Merek
Model
Masukkanbaju ()
Keluarkanbaju ()

Gambar II.12 : Operation pada class diletakkan dibawah attribute

(Sumber : Munawar ; 2005 : 37)

II.8.3 Activity Diagram

Activity Diagram adalah teknik untuk mendeskripsikan logika procedural, proses bisnis dan aliran kerja dalam banyak kasus. Diagram aktifitas mempunyai peran seperti halnya *flowchart*, akan tetapi perbedaannya dengan *flowchart* adalah *activity diagram* bisa mendukung perilaku paralel sedangkan *flowchart* tidak bisa. Contoh *activity diagram* sederhana dapat dilihat pada gambar II.13 sebagai berikut. (Munawar ; 2005 : 109)



Gamabar II.13 : Contoh Activity diagram sederhana

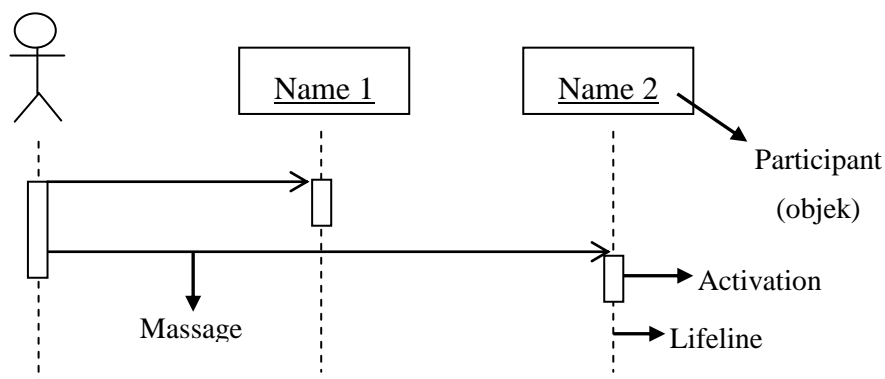
(Sumber : Munawar ; 2005 : 111)

II.8.3 Sequence Diagram

Sequence diagram digunakan untuk menggambarkan perilaku pada sebuah *scenario*. Diagram ini menunjukkan sejumlah contoh obyek dan *message* (pesan) yang diletakkan diantara obyek-obyek ini dalam *use-case*. Komponen utama *Sequence diagram* terdiri atas obyek yang dituliskan dengan kotak segiempat

bernama. *Message* diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan progress vertical. (Munawar ; 2005 ; 87)

Esensi *symbol* dari *sequence diagram* dan *symbol* kerjanya secara bersama-sama. *Participant* terletak di sebelah atas. Setiap *lifeline* menggunakan garis putus-putus yang menurun dari *participant*. Garis yang solid dengan tanda panah menghubungkan antara satu *lifeline* dengan *lifeline* yang lain dan mewakili sebuah *message* dari satu *participant* ke *participant* yang lain. Dari gambar bisa di lihat oseorang *actor* menganalisis *sequence diagram* meskipun *actor* bukan bagian dari *sequence diagram*. Seperti gambar II.14 sebagai berikut :



Gambar II.14 : Simbol-simbol yang ada pada *Sequence diagram*

(Sumber : Munawar ; 2005 : 89)